Deep caching for Agent-based analysis and optimization

Rune Vestvik

***Abstract*—Agent-based modeling and simulation is a "bottom-up" approach for system research, optimization and decision support. The method consist of mapping real world processes into a digital twin model [1] and performing risk-free experiments on the digital twin before applying changes back into real world scenarios.**

The need for computation resources increases as models grow more detailed. This fact, combined with high stochasticity and the use of Monte Carlo simulations, slows down the processes of performing sensitivity analysis and model optimization.

This paper introduces a method for caching agent-based simulations for increased performance and shorting of the feedback loop.

***Index Terms*—deep learning, agent-based modeling, optimization, analysis, deep caching**

I. Introduction

In contrast to analytical modeling, agent-based modeling focuses on the atomic (or minimal) parts of a model. Each agent is modeled individually and set to interact with each other and a given environment (micro scale). The emergent properties of the complete model will then be apparent during simulating (macro scale).

Given randomness in models Monte Carlo simulations are often utilized to explore the full domain of the models [2]. This is typically accomplished by running multiple simulations with different random seed and statistical confidence level for estimates as output for analysis.

Deep neural networks has proved to be a highly useful method for building analytical models based on training data. In the section *Deep caching* algorithm below this is explored as a way to enable deep learning combined with agent-based simulation in order to build a high-responsive analysis of agent-based models.

II. Deep caching algorithm

As complexity increases in agent-based models, the performance slows down. This is both due to the increase in simulation steps per simulation and the requirement for a higher number of Monte Carlo runs in order to retrieve significant results. Sensitivity analysis [3] and parameter optimization is highly impacted and extended to any user experience,

Deep caching is a way of improving the overall responsiveness of the model. This can be implemented by following the steps below:

1. Identify feature vectors to the neural network from the agent-based simulation input parameters.

2. Train neural network with random or swiping input parameters as network input and simulation results as expected output from the network. Exclude invalid and implausible inputs to limit the size of the network.

3. Perform analysis and optimization on the neural network.

4. Verify results in simulation model.

How to optimize the performance of neural networks in general is not covered here. Some degree of experimentation is needed fo find the optimal neural network setup.

III. Conclusion

The approach outlined is applicable in any kind of agent-based simulation model including all domains listen in Table I.

The intersection between agent-based modeling and deep learning has a lot of interesting possibilities. A possible extension to the method described in this paper is to enable use of deep learning to train agents and mechanisms inside of the agent-based simulation using reinforcement learning. [4]

TABLE I
AGENT-BASED MODELING USAGE DOMAINS

IT	Transportation
Aerospace	Rail logistics
Healthcare	Business processes
Markets and competition	Social and ecosystem dynamics
Manufacturing	Education
Supply chains	Energy
Defense	Project and asset management
Pedestrian	Emergency planning
Financial processes	

References

[1] B. S. and R. R, "Boschert S., Rosen R. (2016) Digital Twin—The Simulation Aspect. In: Hehenberger P., Bradley D. (eds) Mechatronic Futures. Springer, Cham," in *Digital Twin—The Simulation Aspect*.

[2] R. Axtell, "WHY AGENTS? ON THE VARIED MOTIVATIONS FOR AGENT COMPUTING IN THE SOCIAL SCIENCES."

[3] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto, "Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models," in *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*.

[4] W. RAND, "MACHINE LEARNING MEETS AGENT-BASED MODELING: WHEN NOT TO GO TO A BAR."

Rune Vestvik (e-mail: rvestvik@gmail.com)